

Amendment
U.S. Application No.: 09/164,777

Claims 1-23 have been rejected under 35 U.S.C. 103(a) as being unpatentable over Misra et al. in view of U.S. Patent No. 5,684,951 to Goldman et al. and U.S. Patent No. 5,479,639 Ewertz et al.

The cited references do not render the present invention obvious as they do not teach or suggest, among other things, storing a verification structure, such as a software license information, in the BIOS of a computer as is recited in the present claims.

Misra et al. is cited as the primary reference against the present claims. Misra relates to a system and method for enforcing software licenses. The system of Misra generates unique identifiers for servers and clients, col 12, lines 41-42. The client system ID 142 is a unique identifier for the client computer, col 12, lines 50-51. The client system IDs can be based on information collected from a computer's hardware and installed software. For example, hard disk volume numbers, registered software, video cards, and some microprocessors contain unique identifiers. This information can be combined to uniquely identify a particular PC. Thus, the client system ID of Misra, is similar to the pseudo-unique key recited in claims 1 and 20.

Misra also describes a license ID, which is a unique identifier assigned to a software license when the software license is issued to a client device, col. 11, lines 9-12. The license ID may be a digital certificate indicating the right to use the particular software at issue, col. 10, lines 60-67. The license ID of Misra is similar to the verification structure and license information recited in claims 1 and 20, respectively.

Misra fails to teach using the BIOS of a computer to store the license ID, as noted in Section 7, Page 6 of the Office Action. Ewertz is cited as supplementing Misra to teach this feature. However, the license information described in Ewertz has a different meaning and a different function from the license information described in Misra. Therefore, a combination of these references would not result in the claimed invention, as is discussed in detail below.

Amendment

U.S. Application No.: 09/164,777

In Ewertz, a "software license number" is described as one type of identification information, col. 3, lines 20-22. This identification information may also include an Ethernet address or system serial numbers, col 3, lines 20-22. The identification information is a unique identification value stored in a non-writable, non-erasable area of the BIOS during manufacture. The identification information uniquely identifies a particular computer. Therefore, according to Ewertz a "software license number" is one of a type of static data structures identifying a specific computer and the static data structure is stored such that it cannot be modified. Accordingly, the software license number of Ewertz is simply identification for the operating system of a particular computer.

For example, col. 2, lines 47-49 of Ewertz disclose that the memory storing the identification information may be electronically locked to prevent erasure or modification of its contents once installed. Moreover, in teaching a preferred embodiment, col. 11, line 23 - col. 12, line 14 of Ewertz describe that several types of identification information must be retained for individual computer systems. One type of identification number, as mentioned above, is an Ethernet address. The Ethernet address is stored in a protected area 306 in static page 2 of the flash memory of Ewertz and cannot be erased or altered once the device is installed. Thus the identification number cannot be destroyed. Ewertz also teaches other computer system identification numbers, such as unique serial number, printed board assembly (PBA) numbers or operating system license numbers may be stored in the locked memory.

Consequently, Ewertz teaches storing identification information for the computer in a non-writable, non-erasable non-volatile memory. This identification information of Ewertz corresponds to the pseudo-unique key stored in the first non-erasable, non-volatile memory as recited in claims 1 and 20 and does not correspond to the license information recited in these claims. The identification information of Ewertz is a static data structure, like the system ID of

Amendment
U.S. Application No.: 09/164,777

Misra, that uniquely identifies a computer and simply does not correspond the license ID of Misra or the license information of the present invention as defined by claims 1 and 20.

From the above discussion, it is clear that the "software license number" according to Ewertz is equivalent in definition and function to Misra's system ID. Therefore, even if Misra is combined with Ewertz, this combination does not result in the present invention. The proposed combination results in the system ID of Misra being stored in the BIOS, not the verification structure or license information being stored in the BIOS as is required by the present claims.

Furthermore, there is no suggestion or motivation to combine Misra and Ewertz in the manner suggested in the Office Action. BIOS is a configuration utility. Software license management applications, such as the one of the present invention, are operating system (OS) level programs. Therefore, BIOS programs and software licensing management applications do not ordinarily interact or communicate because when BIOS is running, the computer is in a configuration mode, hence OS is not running. Thus, BIOS and OS level programs are normally mutually exclusive.

Ewertz teaches that writing to the BIOS area is performed by the BIOS routines:

"Referring to Fig. 8, processing logic for updating the flash memory device with configuration data, such as EISA information, is illustrated... The processing logic shown in Fig. 8 resides in the system BIOS of the preferred embodiment" Col 10, lines 20-28

Misra teaches a licensing system that is OS level based:

"The license generator 26, license server 28 and intermediate server 32 are preferably implemented as computer servers, such as Windows NT servers that run Windows NT server operating systems from Microsoft corporation or UNIX-based servers" Col 5, lines 3-7

Thus, the systems described in Misra and Ewertz are an OS program and a BIOS program, respectively, that cannot run at the same time. Therefore, there is no teaching or suggestion to combine these programs. In fact such a combination would change the operation

C

Amendment
U.S. Application No.: 09/164,777

of the programs, which is an indicia of non-obviousness, see MPEP Sec. 2141.03 and related case law.

Moreover, the present invention proceeds against conventional wisdom in the art. Using BIOS to store application data such as that stored in Misra's local cache for licenses is not obvious. The BIOS area is not considered a storage area for computer applications. An ordinary skilled artisan would not consider the BIOS as a storage medium to preserve application data for at least two reasons.

First, OS does not support this functionality and is not recognized as a hardware device like other peripherals. Every OS provides a set of application program interfaces (APIs) for applications to access storage devices such as hard drives, removable devices, etc. An ordinary person skilled in the art makes use of OS features to write data to storage mediums. There is no OS support whatsoever to write data to the system BIOS. Therefore, an ordinary person skilled in the art would not consider the BIOS as a possible storage medium. Furthermore, it is common that all peripheral devices in the PC are listed and recognized by the OS except for the BIOS. This supports the fact that the BIOS is not considered a peripheral device. Accordingly, an ordinary person skilled in the art would not consider the BIOS for any operation, including writing to the BIOS.

Second, no file system is associated with the BIOS. Every writable device connected to the PC is associated with an OS file system to arrange and manage data structures. An example for such a file system would be FAT, FAT32, NTFS, HPFS, etc. that suggests writing data to the writable device. No such file system is associated with the BIOS. This is further evidence that OS level application programmers would not consider the BIOS as a storage medium for license data.

C

Amendment
U.S. Application No.: 09/164,777

Additionally, Misra teaches away from using the BIOS as a storage area by making a statement about client computers that do not have a persistent non-volatile area.

"The license cache 136 is kept in persisted (non-volatile) storage. Clients that do not have persistent storage can be issued licenses as long as they can generate a unique client ID and can respond to the client platform challenge protocol" (Misra, Col. 12, lines 15-18)

Since all computers must have a BIOS, it is clear Misra teaches away from using the BIOS as a local storage area for licenses.

Goldman et al. do not supplement Misra and Ewertz to teach or suggest the present invention.

Thus, in view of the above discussion, it is clear that the cited references, taken alone or in any combination, do not fairly teach or suggest the present invention. Therefore the withdrawal of this rejection is respectfully requested. Favorable reconsideration of this case and early issuance of a Notice of Allowance is respectfully requested.

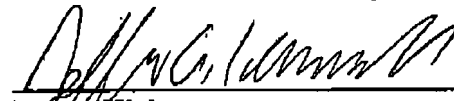
Attached hereto is a marked-up version of the changes made to the specification and claims by the current amendment. The attached page is captioned "Version with markings to show changes made."

In view of the foregoing, reconsideration and allowance of this application are believed in order, and such action is earnestly solicited.

Amendment
U.S. Application No.: 09/164,777

The Commissioner is authorized to charge any fee necessitated by this Amendment to our
Deposit Account No. 22-0261.

Respectfully submitted,



Robert Kinberg
Registration No. 26,924
Jeffri A. Kaminski
Registration No. 42, 709
P.O. Box 34385
Washington, D.C. 20043-9998
Telephone 202-962-4800
Telefax 202-962-8300

RK/JAK/lrh
#347353

Amendment
U.S. Appln. No.: 09/164,777

VERSION WITH MARKINGS TO SHOW CHANGES MADE

IN THE CLAIMS:

Please cancel claims 11, 12, 14 and 15 without prejudice to their re-entry at a later date.

Please amended the claims as follows:

16. (Amended) The method of Claim 221, wherein ~~the~~ a pseudo-unique key ~~includes a pseudo-unique key~~ is stored in the non-volatile memory of the BIOS.

20. (Amended) A method for accessing an application software program using a pseudo-unique key stored in a first non-erasable non-volatile memory area of a computer, the first non-volatile memory area being unable to be programmatically changed, the method, comprising:

loading ~~the~~ an application software program residing in a non-volatile memory area of the computer;

extracting license information from the software program;

encrypting license information using the pseudo-unique key stored in the first non-volatile memory area;

storing the encrypting ~~—~~license information in a second erasable, writable, ~~non~~ non-volatile memory area of the BIOS of the computer;

subsequently verifying the application software program based on the encrypted license information stored in the second erasable, writable, non-volatile memory area of the BIOS; and

acting on the application software program based on the verification.